What is claimed is:

1. A microprocessor for processing instructions, comprising:

a plurality of clusters for receiving the instructions, each of the clusters having a plurality of functional units for executing the instructions; and

a plurality of register sub-files each having a plurality of registers for storing data for executing the instructions,

wherein each of the clusters is associated with corresponding one of the register sub-files so that an instruction dispatched to a cluster is executed by accessing registers in a register sub-file associated with the cluster to which the instruction is dispatched.

2. The microprocessor of claim 1, wherein each of the register sub-files has one write port to which a corresponding cluster sends data to be written into registers in a register sub-file associated with the corresponding cluster.

3. The microprocessor of claim 1, wherein the register sub-files each have a same number of registers.

4. The microprocessor of claim 1, further including a register-renaming unit for renaming target registers in an instruction with registers in a register sub-file associated with a cluster to which the instruction is dispatched.

5. The microprocessor of claim 4, wherein the register-renaming unit identifies a register to be used to store a value named by a target register in the instruction.

6. The microprocessor of claim 4, further including issue-queue units each of which is associated with a corresponding one of the clusters, an issue-queue unit holding instruction renamed by the register-renaming unit until the renamed instruction is issued to be executed in a cluster associated with the issue-queue unit.

7. The microprocessor of claim 6, wherein each of the issue-queue units holds state identifying which instructions need to be executed.

8. The microprocessor of claim 6, further including an instruction dispatch mechanism for determining which of the clusters each instruction is dispatched to.

9. The microprocessor of claim 8, wherein the instruction dispatch mechanism controls the issue-queue units to determine which of the instructions need to be executed.

10. A system for processing an instruction in a microprocessor, comprising:

at least one cluster having at least one functional unit for executing the instruction; and

at least one register file having a predetermined number of physical registers to and from which data is write and read in accordance with the instruction,

wherein the at least one register file has one write port to which an output of the at least one cluster is connected, and data write operation in accordance with the instruction executed by the at least one functional unit is performed by accessing the physical registers of the at least one register file.

11. The system of claim 10, wherein the at least one cluster includes multiple functional units each for executing different instructions.

12. The system of claim 10, further including means for renaming architected registers of the instruction with the physical registers of the at least one register file.

13. The system of claim 12, wherein the architected registers are target registers in which a result of the instruction is stored.

14. The system of claim 12, further including at least one issue-queue unit associated with the at least one cluster, for holding instruction renamed by the

means for renaming until the instruction is issued to be executed in the at least one cluster.

15. A method for processing instructions in a microprocessor, comprising the steps of:

providing clusters each having functional units for executing the instructions;

dividing a register file into a plurality of register sub-files each having registers to store data for executing the instructions;

associating each of the register sub-files with corresponding one of the clusters;

selecting a cluster to which an instruction is dispatched;

renaming target registers in the instruction with registers in a register sub-file associated with the selected cluster; and

dispatching the instruction to the selected cluster wherein the instruction is executed by functional units.

16. The method of claim 15, wherein the dividing step includes assigning a same number of registers to each of the register sub-files.

17. The method of claim 15, wherein the associating step includes providing one write port for each of the register sub-files so that a cluster

associated with a register sub-file sends data to be written to a write port of the register sub-file.

18. The method of claim 15, wherein the renaming step includes identifying a register in a register sub-file to be used to store value named by a target register in the instruction.

19. The method of claim 15, further including holding an instruction renamed in the renaming step until the renamed instruction is issued to be executed by a cluster.